Flexible control of load by using policy-based control of small distributed resources

Kullmann, Daniel

Publication date:
2009

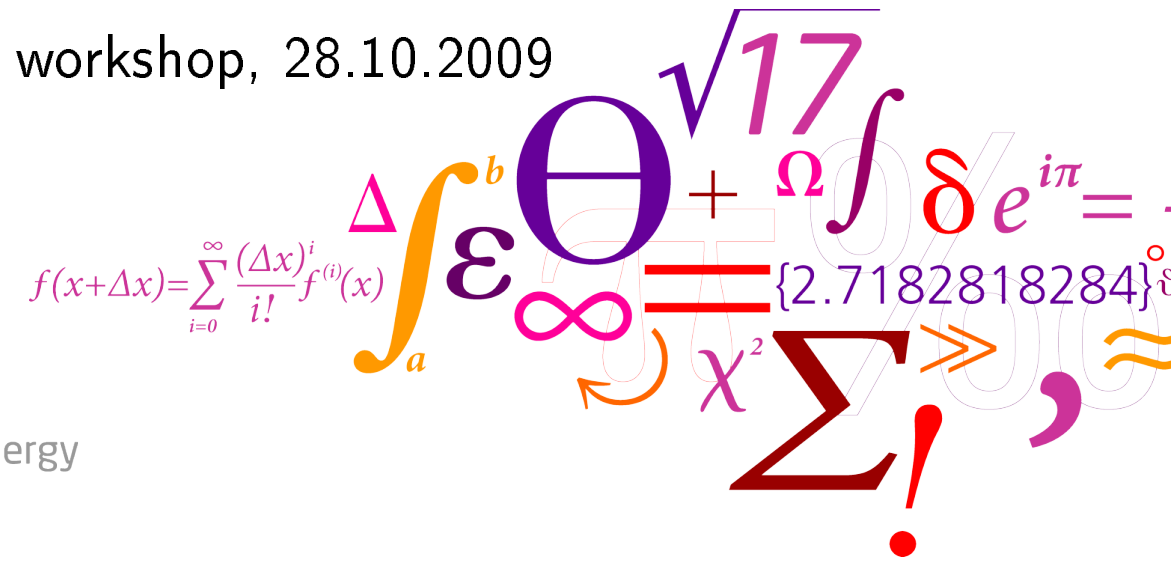
Document Version
Publisher's PDF, also known as Version of record

Policy-based communication for the control of distributed energy systems

Daniel Kullmann

Wind Energy Systems, Risø DTU

2nd VES workshop, 28.10.2009



Risø DTU
National Laboratory for Sustainable Energy

Overview

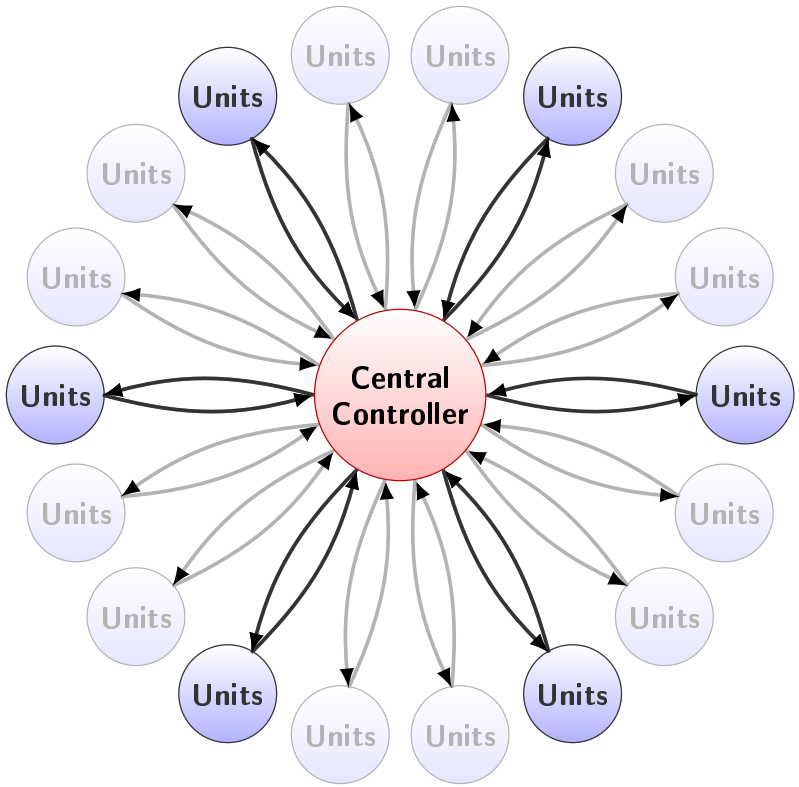
- Issues
- Policy-based control
- ECA rules
- Examples
- Conclusion

Issue: Scalability

- Central controller + many devices: too much load
- Closed-loop control not possible

$\Rightarrow$ Make controlled units more intelligent/autonomous

$\Rightarrow$ Abstract communication necessary



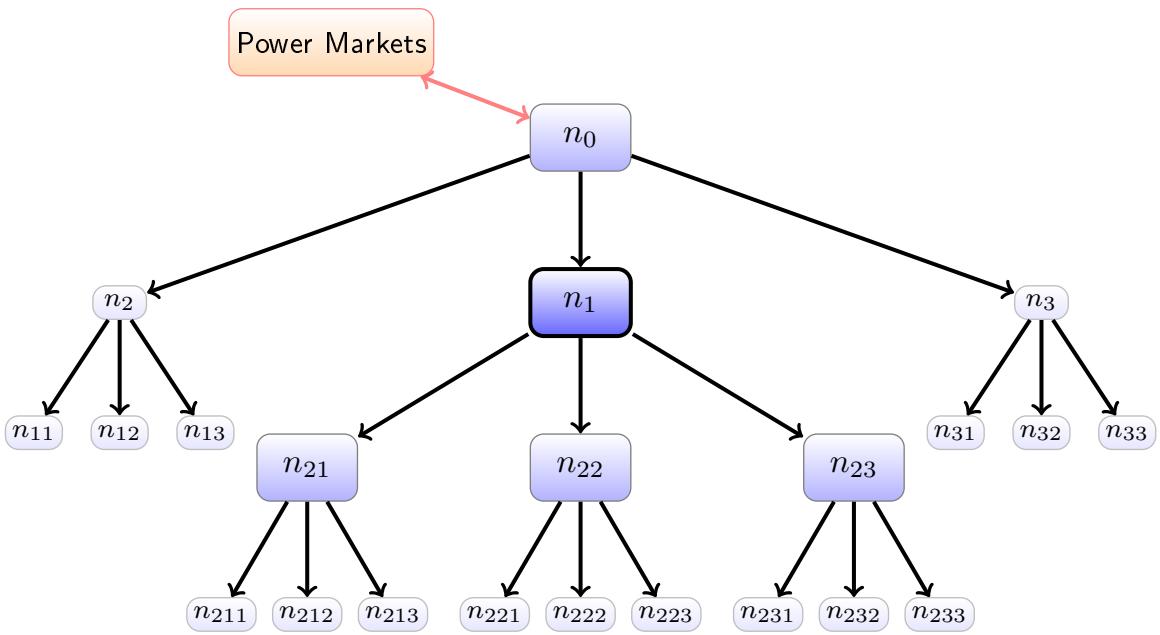
Issue: Aggregation

- Only local information available to nodes
- Aggregation = information is condensed/lost

$\Rightarrow$ Communication protocol necessary

- Support many devices and services

$\Rightarrow$ Unified protocol to aggregators and units



Issue: Unreliable communication

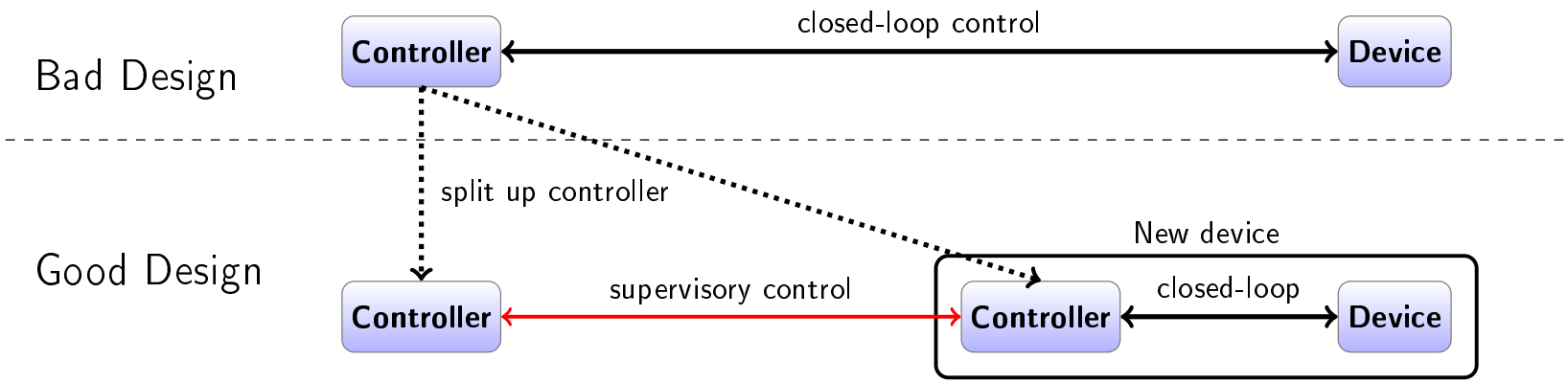
- (Centralised) closed-loop control impossible

$\Rightarrow$ Accept unreliable communication

  - Communication happens when communication works
  - New communication links not necessary$\Rightarrow$low costs

$\Rightarrow$ Move closed-loop control to devices

  - Because closed-loop control needs reliable communication
  - Supervisory control communication necessary



Issues: Reliability & Flexibility

Reliability

- Control must work during problems (e.g. with communication)
- Many devices increase reliability (some may fail)

$\Rightarrow$ Accept unreliable communication

Flexibility

- Flexible behaviour
  - Different ways devices/units can react to changes in environment
- Support different types of devices

  $\Rightarrow$ Generic interfaces for different types of units

Issues: Cost & Services provided

- Keep down costs
  - $\Rightarrow$ Accept unreliable communication
  - $\Rightarrow$ Local control devices not too complex
- Services to be provided
  - Normal operation
  - Emergency operation
  - Ancillary services
  - Other services (e.g. peak shaving)
  - $\Rightarrow$ High-level description of behaviour necessary

Policy-based control

Idea

- Policy: high-level description of behaviour
- Contract that is automatically enforced by units
- Execution of behaviour: independent of communication quality
- Two steps:
  1. Aggregator and unit negotiate behaviour
  2. behaviour is executed
- Units act according to agreed-upon policy
- Units act on external, locally observable events
- Synchronisation, coordinated action

Negotiation protocol

- Negotiation
  1. Aggregator requests info (type, properties, constraints, current status) from unit
  2. Aggregator sends policy to unit
  3. Aggregator waits for ack message
  4. If nack message comes, or no reply at all, aggregator starts over
  5. If ack message comes, aggregator assumes that unit uses policy
- Renegotiation
  - When policy expires
  - When unit has to change its policy because its environment has changed

Advantages & Disadvantages

Advantages

- Deterministic behaviour
- After negotiating, no supervisory communication
- Make lower-level units smarter without giving up control
- Splits up control into closed-loop part and supervisory part
- Policies can deal with different types of units, modes of control
- Good suited to aggregation hierarchies

Disadvantages

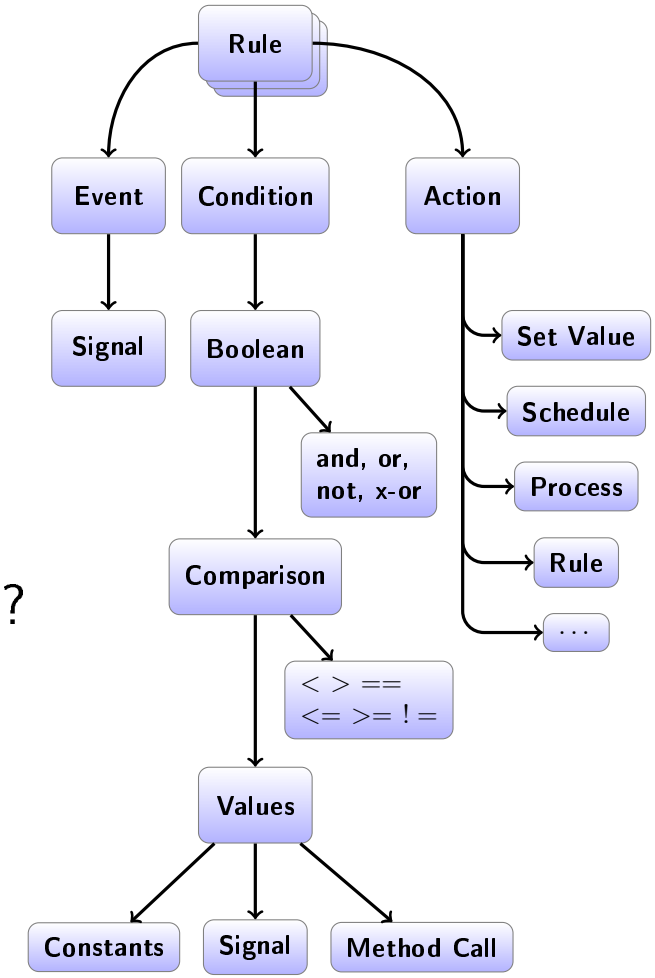
- New communication standard(s) needed
- Possible to have sensible behaviour based on local information?

Policy implementation options

- Implementation ideas:
  - Agents: Autonomous, reactive, proactive, flexible, robust
    - No reasoning from outside possible
    - Aggregator does not know how unit will behave
    - But agents can try different things
  - Send frequency droop curve/schedule/... to unit
    - Limited expressiveness
  - Rule sets expressing the wanted behaviour
    - Flexible, generic policy language needed

ECA Rules as policies

- ECA Policies:
  - Expressive language
  - if-then type of rules
  - Different types of behaviour
- **E**vent:
  - Signal that triggers a rule
- **C**ondition:
  - Current condition interesting to rule?
  - Can rule fix this specific situation?
- **A**ction:
  - Do something about condition
  - Disambiguation: select one of the actions



Example 1: Charging of electric vehicle

Signals:

- Frequency (measured at charging station)
- Power price (broadcast via Internet, GPRS, UMTS, radio)

Rules:

1. frequency < 49.75 Hz $\Rightarrow$ stop charging(if possible)
2. frequency > 50.25 Hz $\Rightarrow$ start charging (if possible)
3. power price $<$ lower threshold $\Rightarrow$ start charging (if possible)
4. power price $>$ upper threshold $\Rightarrow$ stop charging (if possible)

Disambiguation rule:

- Rules 1 and 2 more important than rules 3 and 4

Actions:

- Control charging process

Example 2: Frequency Control with DSM

- Control appliances of many households
- Low frequency: turn off/turn down appliances
- High frequency: turn on/turn up appliances
- Staggering response avoids oscillating behaviour
- Obey characteristics of different appliance types

More Examples

- Tariffs - fixed or schedule
- Frequency based load shedding
- Frequency control
- Real time price signals
- Virtual appliances

Conclusion

Problems

- Only local information: Intelligent behaviour possible?
- Local action, global effect without central oversight?
- Aggregation
- New standard(s) necessary

Further work

- Several use cases
- Agent-based framework
- Experiments in SYSLAB
- Simulating many units
- Input to standardisation